

JUNIT - EXCEPTIONS TEST

http://www.tutorialspoint.com/junit/junit_exceptions_test.htm

Copyright © tutorialspoint.com

JUnit provides a option of tracing the Exception handling of code. You can test the code whether code throws desired exception or not. The **expected** parameter is used along with @Test annotation. Now let's see *@Test(expected)* in action.

Create a Class

- Create a java class to be tested say MessageUtil.java in **C:\ > JUNIT_WORKSPACE**.
- Add a error condition inside printMessage() method.

```
/*
 * This class prints the given message on console.
 */
public class MessageUtil {

    private String message;

    //Constructor
    //@param message to be printed
    public MessageUtil(String message) {
        this.message = message;
    }

    // prints the message
    public void printMessage() {
        System.out.println(message);
        int a =0;
        int b = 1/a;
    }

    // add "Hi!" to the message
    public String salutationMessage() {
        message = "Hi!" + message;
        System.out.println(message);
        return message;
    }
}
```

Create Test Case Class

- Create a java test class say TestJUnit.java.
- Add expected exception ArithmeticException to testPrintMessage() test case.

Create a java class file name TestJUnit.java in **C:\ > JUNIT_WORKSPACE**

```
import org.junit.Test;
import org.junit.Ignore;
import static org.junit.Assert.assertEquals;

public class TestJUnit {

    String message = "Robert";
    MessageUtil messageUtil = new MessageUtil(message);

    @Test(expected = ArithmeticException.class)
    public void testPrintMessage() {
        System.out.println("Inside testPrintMessage()");
        messageUtil.printMessage();
    }
}
```

```

@Test
public void testSalutationMessage() {
    System.out.println("Inside testSalutationMessage()");
    message = "Hi!" + "Robert";
    assertEquals(message,messageUtil.salutationMessage());
}
}

```

Create Test Runner Class

Create a java class file name TestRunner.java in **C:\ > JUNIT_WORKSPACE** to execute Test case(s)

```

import org.junit.runner.JUnitCore;
import org.junit.runner.Result;
import org.junit.runner.notification.Failure;

public class TestRunner {
    public static void main(String[] args) {
        Result result = JUnitCore.runClasses(TestJunit.class);
        for (Failure failure : result.getFailures()) {
            System.out.println(failure.toString());
        }
        System.out.println(result.wasSuccessful());
    }
}

```

Compile the MessageUtil, Test case and Test Runner classes using javac

```
C:\JUNIT_WORKSPACE>javac MessageUtil.java TestJunit.java TestRunner.java
```

Now run the Test Runner which will run test cases defined in provided Test Case class.

```
C:\JUNIT_WORKSPACE>java TestRunner
```

Verify the output. testPrintMessage() test case will be passed.

```

Inside testPrintMessage()
Robert
Inside testSalutationMessage()
Hi!Robert
true

```